M.COM (Computer Management)
W.E.F.2023-24

Program Specific Outcomes

PSO 1	The student will be able to transform complex business scenarios and contemporary issues into problems, investigate, understand and propose integrated solutions using emerging technologies.
PSO 2	Design and development of solutions by applying computer skills, knowledge of quantitative techniques in computer and management applications in practice.
PSO 3	The student will be able to develop a product or process by applying knowledge of programming, web, database, human computer interaction, and networking & security tools.
PSO 4	The student will be able to contribute to research in their chosen field, function, and communicate effectively, to perform both individually and in a multi-disciplinary team.
PSO 5	The student will be able to make decisions related to work that demonstrate intellectual curiosity, a commitment to lifelong learning in students and understanding of being an ethical computing professional with societal and environmental concerns.

Teaching and Examination Scheme

A teaching and examination scheme for students admitted to the M. Com. (Computer Management) Program shall be as follows:

Master of Commerce (Computer Management)
Semester I

Sr. No.	Course Type	Subjects	Course Code	Teaching Scheme			Examination Scheme					Credits
				Total Hours Per Week			Max. Marks SEE (TH) *	Max Marks SEE (PR)	Max. Marks (CIE)	Total Marks	Min. Passing Marks	
				Theory	Practical	Total						
1.	Core	Python Programming		4	-	4	80	-	20	100	40	4
2.	Core	Practical Python Programming		-	8	8	-	100	-	100	50	4
3.	Core	Cloud Computing		4	-	4	80	-	20	100	40	4
4.	Elective	Practical Advance Java / Practical React JS		-	8	8	-	100	-	100	50	4
5.	Core	Research Methodology		4	-	4	80	-	20	100	40	4
				12	16	28	240	200	60	500	**250**	20

* Semester End Examination which is mandatorily required to be appeared by every student

Note:

1. TH = Theory, CIE= Continuous Internal Evaluation
2. SEE for Theory as well as Practical examinations as mentioned above shall be conducted by the University for all semesters and the CIE shall be conducted by colleges on behalf of the University for all Semesters.

Master of Commerce (Computer Management)
Semester II

Sr. No.	Course Type	Subjects	Course Code	Teaching Scheme			Examination Scheme					Credits
				Total Hours Per Week			Max. Marks SEE (TH) *	Max Marks SEE (PR)	Max. Marks (CIE)	Total Marks	Min. Passing Marks	
				Theory	Practical	Total						
1.	Core	ASP.Net		4	-	4	80	-	20	100	40	4
2.	Core	Practical ASP.Net		-	8	8	-	100		100	50	4
3.	Core	Information Security & Cyber Law		4	-	4	80	-	20	100	40	4
4.	Elective	Practical Android Programming / Practical Angular JS		-	8	8	-	100		100	50	4
5.	Core	On Job Training		-	8	8	-	100	-	100	50	4
				8	24	32	160	300	40	500	**250**	20

* Semester End Examination which is mandatorily required to be appeared by every student

Note:

1. TH = Theory, CIE= Continuous Internal Evaluation
2. SEE for Theory as well as Practical examinations as mentioned above shall be conducted by the University for all semesters and the CIE shall be conducted by colleges on behalf of the University for all Semesters.

Master of Commerce (Computer Management)
Semester III

Sr. No.	Course Type	Subjects	Course Code	Teaching Scheme	Examination Scheme				Credits
				Total Hours Per Week	Max. Marks	Max Marks		Total Marks	

				Theory	Practical	Total	SEE (TH) *	SEE (PR)	Max. Marks (CIE)		Min. Passing Marks	
1.	Core	Advance Database Management System		4	-	4	80	-	20	100	40	4
2.	Core	Practical – SQL & PL/SQL		-	8	8	-	100		100	50	4
3.	Core	Management Information System		4	-	4	80	-	20	100	40	4
4.	Elective	Data Communication & Computer Network / Intelligent System (AI)		4	-	4	80	-	20	100	40	4
5.	Core	Research Project		-	8	8	-	100	-	100	50	4
				12	16	28	240	200	60	500	**250**	20

* Semester End Examination which is mandatorily required to be appeared by every student

Note:

1. TH = Theory, CIE= Continuous Internal Evaluation
2. SEE for Theory as well as Practical examinations as mentioned above shall be conducted by the University for all semesters and the CIE shall be conducted by colleges on behalf of the University for all Semesters.

Master of Commerce (Computer Management)
Semester IV

Sr. No.	Course Type	Subjects	Course Code	Teaching Scheme			Examination Scheme					Credits
				Total Hours Per Week			Max. Marks SEE (TH) *	Max Marks SEE (PR)	Max. Marks (CIE)	Total Marks	Min. Passing Marks	
				Theory	Practical	Total						
1.	Core	Software Engineering		4	-	4	80	-	20	100	40	4
2.	Core	Mobile Computing		4	-	4	80	-	20	100	40	4
3.	Core	Big Data & Hadoop		4	-	4	80	-	20	100	40	4
4.	Elective	Practical Ruby on Rail / Practical Web with Word Press		-	8	8	-	100	-	100	50	4
5.	Core	Research Project		-	12	12	-	100	-	100	50	6
				12	20	32	240	200	60	500	**250**	22

* Semester End Examination which is mandatorily required to be appeared by every student

Note:

1. TH = Theory, CIE= Continuous Internal Evaluation
2. SEE for Theory as well as Practical examinations as mentioned above shall be conducted by the University for all semesters and the CIE shall be conducted by colleges on behalf of the University for all Semesters.

QUESTION PAPER PATTERN

First / Second / Third / Fourth Semester

Master of Commerce (Computer Management) - M.Com.(CM)

OB & CBCS Examination

Time: 3 Hours **Total Marks: 80**

N. B. - a) Draw well labeled diagram wherever necessary.

b) All questions are compulsory.

Q1. **8 x 2 = 16**

N. B. – 1. Each question carries two marks.

2. Answers should not more than five lines.

A. Unit I

B. Unit I

C. Unit II

D. Unit II

E. Unit III

F. Unit III

G. Unit IV

H. Unit IV

Q2. **8 x 3 = 24**

N. B. – 1. Each question carries three marks.

2. Answers should not more than ten lines.

A. Unit I

B. Unit I

C. Unit II

D. Unit II

E. Unit III

F. Unit III

G. Unit IV

H. Unit IV

N. B. – 1. Each question carries five or ten marks.

2. Answers should not more than 250 words for 5 marks questions and 600 words for 10 Marks questions respectively.

Q3. **Either**

(A) 5 Unit I

(B) 5 Unit I

OR

(C) 10 Unit I

Q4. **Either**

(A) 5 Unit II

(B) 5 Unit II

OR

(C) 10 Unit II

Q5. **Either**

(A) 5 Unit III

(B) 5 Unit III

OR

(C) 10 Unit III

Q6. **Either**

(A) 5 Unit IV

(B) 5 Unit IV

OR

(C) 10 Unit IV

Master of Commerce (Computer Management) – M.Com(CM)

Semester – I

Paper – 1

Course Code – **Course Name – Python**

Learning Outcome

LO1	Given information on different types of programming languages so that Students will be able to **distinguish** the high-level language and **understand** the benefits of using python for development of application program.
LO2	Given information on control statements of program student will be able to understand the program flow and will able to **implement** various control statement and functions for effective code design.
LO3	Given information on advance program structure Students will able to **interpret** multiple data structured elements while developing real life application for business solution.
LO4	Given information on basics of object oriented programming student will be able **create and use** different types of objects, classes and File handling operations for redesigning the program structure.

UNIT - I

The Way of the Program - The Python Programming Language, What Is a Program?, What Is Debugging?, Syntax Errors, Runtime Errors, Semantic Errors, Experimental Debugging, Formal and Natural Languages, The First Program. **Variables, Expressions, and Statements** - Values and Types, Variables, Variable Names and Keywords, Operators and Operands, Expressions and Statements, Interactive Mode and Script Mode, Order of Operations, String Operations, Comments. **Functions** - Function Calls, Type Conversion Functions, Math Functions, Composition, Adding New Functions, Definitions and Uses, Flow of Execution,

Parameters and Arguments, Variables and Parameters Are Local, Stack Diagrams, Fruitful Functions and Void Functions, Why Functions?, Importing withfrom.

UNIT - II

Conditionals and Recursion - Modulus Operator, Boolean Expressions, Logical Operators, Conditional Execution, Alternative Execution, Chained Conditionals, Nested Conditionals, Recursion, Stack Diagrams for Recursive Functions, Infinite Recursion, Keyboard Input. **Fruitful Functions** - Return Values, Incremental Development, Composition, Boolean Functions, More Recursion, Leap of Faith, One More Example, Checking Types. **Iteration** - Multiple Assignment, Updating Variables, The while Statement, break, Square Roots, Algorithms, Debugging. **Strings** - A String Is a Sequence, len, Traversal with a for Loop, String Slices, Strings Are Immutable, Searching, Looping and Counting, String Methods, The in Operator, String Comparison.

UNIT - III

Lists - A List Is a Sequence, Lists Are Mutable, Traversing a List, List Operations, List Slices, List Methods, Map, Filter, and Reduce, Deleting Elements, Lists and Strings, Objects and Values, Aliasing, List Arguments. **Dictionaries** - Dictionary as a Set of Counters, Looping and Dictionaries, Reverse Lookup, Dictionaries and Lists, Memos, Global Variables, Long Integers. **Tuples** - Tuples Are Immutable, Tuple Assignment, Tuples as Return Values, Variable-Length Argument Tuples, Lists and Tuples, Dictionaries and Tuples, Comparing Tuples, Sequences of Sequences.

UNIT - IV

Files – Persistence, Reading and Writing, Format Operator, Filenames and Paths, Catching Exceptions, Databases, Pickling, Pipes, Writing Modules. **Classes and Objects** - User-Defined Types, Attributes, Rectangles, Instances as Return Values, Objects Are Mutable, Copying. **Classes and Functions** – Time, Pure Functions, Modifiers, Prototyping Versus Planning. **Classes and Methods** - Object-Oriented Features, Printing Objects, Another Example, A More

Complicated Example, Theinit Method, The str Method Operator Overloading, Type-Based Dispatch, Polymorphism, Debugging, Interface and Implementation. **Inheritance** - Card Objects, Class Attributes, Comparing Cards, Decks, Printing the Deck, Add, Remove, Shuffle, and Sort, Inheritance, Class Diagrams, Debugging, Data Encapsulation.

Text Book:

1. Allen B. Downey, Think Python, Shroff Publishers, O'Reilly.

Reference Books:

1. Charles Dierbach, Introduction to Computer Science using Python,Wiley.
2. Laura Cassell & Alan Gauld, Python Projects, Wrox A WileyBrand.
3. Paul Greis, Jennifer Campbell, Jason Montojo, Practical Programming – An Introduction to Computer Science using Python, ShroffPublishers.

Practical List of Python

1. Write a Python program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon users choice.
2. Write a Python program that allows the user to enter any integer base and integer exponent, and displays the value of the base raised to that exponent.
3. Write a Python program to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria:

   Grade A: Percentage >=80

   Grade B: Percentage>=70 and <80

   Grade C: Percentage>=60 and <70

   Grade D: Percentage>=40 and <60

   Grade E: Percentage<40

4. Write a Python program to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user using user-defined function.
5. Write a Python program to swap two variables
6. Write a Python program to input any 10 numbers and calculate their average using user defined function?
7. Write a Python Program to Check if a Number is Positive, Negative or 0
8. Write a Python Program to Check if a Number is Odd or Even
9. Write a Python program to find the largest number among the three input numbers
10. Write a Python Program to Display the multiplication Table
11. Write a Python program to print current date and time. In addition to this, print each component of date(i.e.year,month,day) and time (i.e.hours, minutes and microseconds) separately.
12. Write a Python program to calculate the subtraction of two compatible matrices ?

13. Write a Python program to print first 10 prime numbers.
14. Write a Python Program to Check Whether a String is Palindrome or Not
15. Write a Python Program to Convert Decimal to Binary, Octal and Hexadecimal
16. Write a Python Program to Sort Words in Alphabetic Order.
17. Write a Python Program to Find Sum of Natural Numbers Using Recursion
18. Write a Python Program to Print the Fibonacci sequence
19. Write a Python Program to Convert Kilometers to Miles
20. Write a Python Program to Find the Square Root

Paper – II

Course Code - **Course Name: Cloud Computing**

Learning Outcome

LO1	Given information on basics of Cloud Computing, students will be able to **understand** the different paradigms, also able to **define** fundamental terminologies of Cloud Computing.
LO2	Given information on architecture and deployment model, students will be able to **remember** structure and the use of Cloud management with its types.
LO3	Given information on various cloud service models students will be able to **differentiate** and illustrate its Virtualization.
LO4	Given information on service providers, students will be able to **describe** the cloud service Model and understand the importance of its service providers.

UNIT-I

Computing Paradigms: High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Bio computing, Mobile Computing, Quantum Computing, Optical Computing, Nano computing, Network Computing. **Cloud Computing Fundamentals**: Need for Cloud Computing, Defining Cloud Computing; Network Computing Cloud Computing as a Service, Cloud Computing as a Platform, Principles of Cloud computing, Essential Characteristics, Requirements for cloud services, Benefits and Drawbacks of cloud computing.

UNIT-II

Cloud Computing Architecture and Management: Cloud architecture, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Cloud Applications, Managing the Cloud-Cloud Infrastructure and Cloud application, Migrating Application to Cloud, Phases of Cloud Migration, Approaches for Cloud Migration, **Cloud deployment model-** Characteristics, Types of Deployment -Private, Public, community and hybrid.

UNIT-III

Cloud Service Models: Infrastructure as a Service, **IaaS-** Characteristics, Suitability, Pros and Cons, Summary of IaaS Providers, Platform as a Service **PaaS-** Characteristics, Suitability, Pros and Cons, Summary of PaaS Providers, Platform as a Service, Software as a Service. **SaaS** - Characteristics, Suitability, Pros and Cons, Summary of SaaS Providers, Platform as a Service, Other Cloud Service Models. **Virtualization:** Opportunities, Approaches to virtualization.

UNIT-IV

Cloud Service Providers: EMC, EMC IT, Captiva Cloud Toolkit, Google, Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue, service, Microsoft, Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM, Cloud Models, IBM Smart Cloud, SAP Labs, SAPHANA Cloud Platform, Virtualization Services Provided by SAP, Sales force, Sales Cloud. **Service Cloud:** Knowledge as a Service, Rack space, VMware, Manjra soft, Aneka Platform.

Text Book:

1. Essentials of cloud Computing: K. Chandrasekhran, CRCpress,2014

Reference Book:

1. Cloud Computing: Principles and Paradigms by Rajkumar Buyya,James Broberg and Andrzej M. Goscinski, Wiley, 2011.
2. Cloud Computing: A Practical Approach, Anthony T.Velte, Toby J.Velte, Robert Elsenpeter, Tata McGraw Hill, 2011.
3. Cloud Computing: Implementation, Management and Security, John W. Rittinghouse, James F.Ransome, CRC Press, 2012.

Paper – III

Course Code **Course Name – Advance Java**

Learning Outcome

LO1	Given information on the use of connectivity (JDBC) and networking which helps for client server application, students will be able to **create** management applications practices emphasized for network based client server application.
LO2	Given information is used for creation of enterprise edition work with servlet's and session tracking mechanism; students will be able to **develop** the solution for human computer interaction.
LO3	Given information on event handling and Java Server Pages, students will able to **design** and **create** the web by using action tags with the help of JSP API
LO4	Given information on Extensions and Standard Tags library students will be able to Apply advance Tags in their web pages and able to **design** and **develop** the application by using technologies.

UNIT - I

Introducing Swing – JFC, The MVC Architecture, Applet, Window Panes, Important Classes of the javax.swing Package, Setting the Look and Feel of Components, An Applet with Swing Components. **Working with JDBC** - Introducing JDBC, Exploring JDBC Drivers, Exploring the Features of JDBC, Describing JDBC APIs, Exploring Major Classes and Interfaces, Exploring JDBC Processes with the java.sql Package, Working with Transactions. **Network Programming** - Networking Basics, Network Programming in Java Using the java.net Package, Establishing the two-way Communication between Server and Client, Retrieving a file at server, Learning the DatagramSocket and DatagramPacket Classes, Understanding the Content and Protocol Handlers.

UNIT - II

RMI, Naming Service, Serialization, and Internationalization - RMI Architecture, RMI Registry, Dynamic Code Loading in RMI, RMI API, Creating a Distributed Application, using RMI, Naming Services, Directory and Naming Services, Overview of JNDI, Object Serialization, Internationalization, Java and Internationalization, Internationalizing Web Applications. **Introducing the Java EE Platform** - Enterprise Application Concepts, Introducing the Java EE 6 Platform, HTTP Protocol, Exploring Web Application, Introducing Web and Application Servers. **Working with Servlets** - Exploring the Features of Java Servlet, Exploring New Features in Servlet 3.0, Exploring the Servlet API, Explaining the Servlet Life

Cycle, Understanding Servlet Configuration, Creating a Sample Servlet, Creating a Servlet by using Annotation, Working with ServletConfig and ServletContext Objects, Working with the HttpServletRequest and HttpServletResponse Interfaces, Exploring Request Delegation and Request Scope, Describing a Session, Introducing Session Tracking, Exploring the Session Tracking Mechanisms, Using the Java Servlet API for Session Tracking.

UNIT - III

Introducing Event Handling and Filters - Introducing Events, Intoducing Event Handling, Working with the Types of Servlet Events, Introducing Filters, Exploring Filter API, Configuring a Filter, Creating a Web Application Using Filters, Using Initializing Parameter in Filters, Manipulating Responses, Discussing Issues in Using Threads with Filters. **Working with JavaServer Pages (JSP)** - Introducing JSP Technology, Exploring New Features of JSP 2.1, Listing Advantages of JSP over Java Servlet, Exploring the Architecture of a JSP Page, Describing the Life Cycle of a JSP Page, Working with JSP Basic Tags and Implicit Objects, Working with Action Tags in JSP, Exploring the JSP Unified EL, Using Functions with EL.

UNIT - IV

JSP Tag Extensions and Standard Tag Library - Exploring the Elements of Tag Extensions, Exploring the Tag Extension API, Working with Classic Tag Handlers, Working with Simple Tag Handlers, Working with JSP Fragments, Working with Tag Files, Introducing JSTL, Working with the Core Tag Library, Working with the XML Tag Library, Working with the Internationalization Tag Library, Working with the SQL Tag Library, Working with the Functions Tag Library. **Introducing Hibernate** - Introducing Hibernate, Exploring the Architecture of Hibernate, Downloading Hibernate, Exploring HQL, Understanding Hibernate O/R Mapping, Working with Hibernate, Implementing O/R Mapping withHibernate.

Text Book:

1. Prof. M. T. Savaliya, Advance java Technology, DreamtechPress.

Reference Books:

1. Dr. Ashwin Mehta, Sarika Shah, Advance Java for Students, ShroffPublishers.
2. Patrick Naughton & Herbert Schildt, The Complete Reference: Java 2, McGraw- Hill.
3. Joseph Weber, Using Java 2 Platform, Prentice Hall ofIndia.

4. Uttam K. Roy, Advance Java Programming, OxfordUniversity.
5. Kanika Lakhani, Advance Java Programming, KatsonBooks.

Practical List of Advance Java

1. Write a Java program to develop an applet that draws a circle. The dimension of the applet should be 500 x 300 pixels. The circle should be centered in the applet and have a radius of 100 pixels. Display your name centered in a circle.(using drawOval() method).
2. WriteaJavaprogramtodrawtenredcirclesinaverticalcolumninthecenterofthe applet.
3. Write a Java program to develop calculator-using Swing and add image on Button.
4. Write a Java program to find the IP address or computer name of local machine.
5. Write a Java program with class Greeting Client is a client program that connects to a server by using a socket and sends a greeting, and then waits for a response.
6. Write a Java program that implements a simple client/server application. The client sends data to a server the server receives the data, uses it to produce a result and then sends the result back to the client. The client displays the result on the console. For ex the data send from the client is a numbers and the result produce by the server is the addition of that number.
7. Write a Java program to create an application that displays a frame with a menu bar. When a user selects any menu or menu item, display that selection on a text area in the center of the frame.
8. Write RMI application where client supplies data to withdraw and server response with new account balance. Provide your custom security policy for this application.
9. Write a Java program to develop database application that allows user to Insert, Update, Delete values in a Table and manages appropriate exception handling when wrong values are enter.
10. WriteaJavaprogramtopresentasetofchoiceforusertoselectaproductand display the price of product.
11. Write a Java program to show validation of user using servlet.
12. Write a Java program to develop a simple servlet question answer application
13. Write a Java program to pass any URL string and display all 4 elements of URL string.
14. Write a Java program to trap all the events of mouse listener interface.

15. Write a Java program to show validation of user using JSP.
16. Write a Java program to display message on browser using JSP.
17. Write a Java program to connect with the google.com and retrieve the html code of default webpage.
18. Write a Java program to present a set of choices for a user to select stationary products and display the price of product after selection from the list.
19. Write a Java program to demonstrate typical editable table, describing employee details for a software company.
20. Write a Java program to trap all the events of key listener interface.
21. Write a Java program of calling one servlet by another servlet.
22. Write a Java program to develop a simple servlet calculator application.
23. Write a Java program to set scope of beans.
24. Write a Java program to create a JSP application that accepts registration details from the student and stores the details into the database table.
25. Write a Java program to develop a JSP application that authenticate user login as per the registration details. If login success then forward user to the index page otherwise show login failure message.
26. Write a Java program using split pane to demonstrate a screen divided into two parts contains a name of planets and another display the image of planet. When user selects the planet name from the left screen appropriate image of display in right screen.
27. Write a Java program to develop a web application to add items in the inventory using JSP.
28. Write a Java program to create a web Form, which processes servlet and demonstrates use of cookies and sessions.
29. Write a Java program to develop a simple JSP program for user login form with static and dynamic database.
30. Write a Java program to develop a JSP program to display the grade of a student by accepting the marks of five subjects.

Paper – III

Course Code **Course Name – React JS**

Learning Outcome

LO1	Given information on ECMS script student will be able to **understand** the basic concept of scripting language as well as able to **use** functions in the program
LO2	Given information on React JS basic concepts students will be able to remember the directory structure and nested elements of JSX
LO3	Given information on props & state, components and routing student will be able to **create** components and **apply** them in their program.
LO4	Given information and form & user input and hooks students will be able **design** Form, **create** and **apply** custom hooks in their application.

UNIT – I

Introduction of ECMS script: Introduction var, let & const, Arrow function, setTimeout & clearTimeout method, setInterval & clearInterval mthods, map function, filter function, join function, callback function, spread operator, reduce function, sort function, sort function, classes, properties and method, import and export

UNIT – II

Introduction of React JS: About ReactJs, server requirements, node Js & NPM, Webpack, Babel & JSX, Directory Structure; **Introduction of JSX**- Basic Concept, nested elements in JSX, JSX attributes, JSX Comments

UNIT – III

Props & States: Introduction, default props, props types, basic states, common Antipattern, SetState, State, Events & Managed controls; **Components**: Class components, functional components; **Routing**: Installation of React Routing, Create components, Add a route

UNIT – IV

Forms and User Input: controlled components, uncontrolled components; **Hooks**: UseState & State updating, multiple states, rules of Hooks, useEffect, custom Hook

Reference Books

1. Learning React: Modern Patterns for Developing React Apps 2nd Edition by Alex Banks & Eve Porcello

2. React Key Concepts: Consolidate your knowledge of React's core features by Maximilian Schwarzmuller
3. The Road to React: Your journey to master plain yet pragmatic React.js by Robin Wieruch
4. React and React Native: Build cross-platform JavaScript applications with native power for the web, desktop, and mobile, 4th Edition 4th ed. Edition by Adam Boduch, Roy Derks and Mikhail Sakhniuk
5. React.js Complete Guide To Server-Side Rendering (Front-end development) by Gerard van der Put
6. Learn React Hooks: Build and refactor modern React.js applications using Hooks by Daniel Bugl

Practical List React JS

1. Write a program to Build Search filter in React - React code to build a simple search filter functionality to display a filtered list based on the search query entered by the user.
2. Write a program to create Simple counter exercise - Creating a simple counter using React which increments or decrements count dynamically on-screen as the user clicks on the button. This exercise requires knowledge of fundamental React concepts such as State, Component, etc.
3. Write a program to Display a list in React - React code to print each item from the list on the page using Array.map() function to display each item on the page.
4. Write a program to Build Accordion in React - Creating an accordion that toggles text content on click of the accordion header using React State and conditional rendering.
5. Write a program to create Image Slider using React JS - React exercise to create an image slide, where users can view multiple images with next/previous buttons. Additionally, there is also an option to select an image from any index of the list through a click-on option circle.
6. Write a program to Create a Checklist in React - React code to display a checklist with multiple options that can select and the selected options are dynamically displayed on the screen. React State is used to keep track of checked options and onChange() Event handler is triggered to alter the state whenever an option is checked or unchecked.

7. Write a program to create Simple Login form in React - React code for simple login form where the user login by entering their username and password. The form inputs are validated to check if correct information is entered and the error messages are the validation fails. The login form is hidden and the "Welcome, ${name}" message is shown when the user login is successful.
8. Write a program to Print data from REST API - React code to collect data from rest API using fetch() in JavaScript combined with useEffect() to load the content on page render.
9. Write a program to create a Multi-Page navigation using React Router - React code to develop a multipage application with navigation for Home, About and Blog pages. The route-based component rendering is implemented using the "react-dom" npm package to allow users to navigate to different pages and render the component with respect to the route.
10. Write a program to create Context API in React Components - Context allows values to be passed from multiple levels of child components without using props. Thus context can be used as an alternative to Redux in some of the cases.
11. Write a program to Create simple Calculator: Create Simple calculator in React
12. Write a program to Create Image Search: Create image search using ReactJs and Unsplash Developer API.
13. Write a program to create Youtube Search: Youtube Video Search in React
14. Write a program to Create Todo List: Create Todo List in React.
15. Write a program to Integrate Bootstrap framework with React and Create Simple Registeration Form
16. Write a program to Integrate Material UI framework with React and Create Simple Registeration Form
17. Create React Fully Responsive Website: Build a fully responsive website in ReactJS, viewable in desktop, mobile, tablet.
18. Create Static Website: Create a static website by creating reusable components like button, cards, container, icons
19. Create Weather App: Create Weather App by using free OpenWeather.
20. Create React Password Generator: Generate strong password using ReactJS.

Semester – II

Paper – 1

Course Code – **Course Name – ASP.Net**

Learning Outcome

LO1	Given information on **development** and **deployment** cycles of enterprise applications so that Students will be able to understand the ASP.NET frame work to and enhance the web page with the combination of advance web designing tools(CSS3, HTML5)build distributed enterprise application
LO2	Given information to **understand** server controls like secure protocols and also **examine** the entered data on the web page which helps to handle Master page with cookies.
LO3	Given information to **access** the backend (database)with suitable connectivity controls and **deploy** a secure client server in real life application with customized web page like secure web access methods
LO4	Given information will **deploy** the web application by application interface control and WCF services so that Students will be able to **create** dynamic web applications using a combination of client-side (JavaScript, HTML, XML, WML) and server-side technologies (ASP.NET, ADO.NET).

UNIT – I

An introduction to ASP.NET programming: An introduction to web applications, An introduction to ASP.NET development. **How to develop a one-page web application**: How to work with ASP.NET web sites, How to use Visual Studio to build a web form, How to add validation controls to a form, How to add C# code to a form, How to test a web application.

How to use HTML5 and CSS3 with ASP.NET applications: The Future Value application with CSS formatting, The HTML and CSS skills that you need. **How to develop a multi-page web application**: How to work with multi-page web sites, How to use session state. **How to test and debug ASP.NET applications**: How to test an ASP.NET web site, How to use the debugger, How to use the trace feature.

UNIT –II

How to use the standard server controls: How to use the common server controls, How to use the button controls, How to use the list controls. **How to use the validation controls**: Introduction to the validation controls, How to use the validators, Validation techniques. **How to work with state, cookies, and URL encoding**: How to use view state, How to use session state, How to use application state and caching, How to use cookies and URL encoding. **How to use master pages**: How to create master pages, How to create and develop content pages, How to customize content pages. **How to use themes**: An introduction to themes, How to work with themes and skins. **How to use site navigation and ASP.NET routing**: How to use the navigation controls, How to use ASP.NET routing, How to use the navigation controls with ASP.NET routing.

UNIT – III

An introduction to database programming: An introduction to relational databases, An introduction to ADO.NET 4.5, How to use the DataList control, How to use data binding, How to customize the GridView control, How to use the DetailsView control , How to use the FormView control. **How to use object data sources with ADO.NET**: An introduction to object data sources, How to create a data access class, A Category Maintenance application . **How to secure a web site**: An introduction to SSL, How to use a secure connection. **How to authenticate and authorize users**: An introduction to authentication, How to set up authentication and authorization, How to use the login controls. **How to use email, custom error pages, and back-button control**: How to send email, How to use custom error handling, How to handle the back-button problem.

UNIT – IV

How to configure and deploy ASP.NET applications: How to use the Web Site Administration Tool, An introduction to deployment, How to use one-click deployment, How to create and use a Setup program. **How to use ASP.NET Ajax**: An introduction to Ajax, An introduction to ASP.NET Ajax, How to use the ASP.NET Ajax server controls, An application that uses ASP.NET Ajax. **How to create and use WCF and Web API services**: An introduction to web services, How to create a WCF service, How to create a web site that consumes a WCF service, How to create a Web API service, How to create a web site that consumes a Web API service. **An introduction to ASP.NET MVC**: An introduction to MVC, An introduction to ASP.NET MVC, How to work with views, How to work with controls and postbacks.

Text Book:

1. Mary Delamater & Anne Boehm, murach's ASP.Net Web Programming with C#, Shroff Publishers.

Reference Books:

1. ASP.Net Black Book, Kogent Learning Solutions Inc, Dreamtech Press.
2. Jason Gaylord, Christian Wenz, Pranav Rastogi, Todd Miranda, Scott Hanselman, Professional ASP.Net in C# & VB, Wrox A Wiley Brand.
3. ASP.Net with C#, Kogent Learning Solutions Inc, Dreamtech Press.

Practical List of ASP.Net

1. Create a page in ASP.NET using VB.NET or C# to display the following Web Controls:
   - A button with text —click mel. The button control must be in the center of the form.
   - A label with a text hello
   - A checkbox. The form name must be Web Controls.
2. Create a page in ASP.NET using VB.NET or C# containing the following controls:
   - A ListBox
   - A Button
   - An Image
   - A Label

The listbox is used to list items available in a store. When the user clicks on an item in the listbox, its image is displayed in the image control. When the user clicks the button, the cost of the selected item is displayed in the control.

3. Create a page in ASP.NET using VB.NET or C# that take a student name from the user, add that name in list-box control. And delete the chosen name from the list-box.
4. Create a page in ASP.NET using VB.NET or C# for book sales. Enter the quantity, title and price of the book. Calculate the extended price, discount (15%) and after discount, the actual price of the book. Show the summery of book sales. (Like total no of books, total discount given, total discounted amount and average discount.) You will need command buttons- calculate, clear sale.
5. Create a page in ASP.NET using VB.NET or C# using HTML Server controls that take user name, address, and city, state and country name from the user and display it.
6. Create a page in ASP.NET using VB.NET or C# to get a user input such as the boiling point of water and test it to the appropriate value using Compare Validator
7. Create a page in ASP.NET using VB.NET or C# that uses a textbox for a user input name and validate it for RequiredField Validation.
8. Create a page in ASP.NET using VB.NET or C# that gets user input such as the user name, mode of payment, appropriate credit card. After the user enters the appropriate values the Validation button must validates the values entered.
9. Create a page in ASP.NET using VB.NET or C# to declare one TextBox control, one Button control, one Label control, and one RegularExpressionValidator control in an .aspx file. The submit() function checks if the page is valid. If it is valid, it returns "The page is valid!" in the Label control. If it is not Valid, it returns "The page is not valid!" in the Label control. If validation fails, the text "The zip code must be 5 numeric digits!" will be displayed in the RegularExpressionValidator control.
10. Create a page in ASP.NET using VB.NET or C# using HTML Server controls that convert given currency into another selected currency. For that you need a dropdown-list.
11. Create a page in ASP.NET using VB.NET or C# to the database with ADO.NET for Inserting Data.
12. Create a page in ASP.NET using VB.NET or C# to the database with ADO.NET for Updating Data.

13. Create a page in ASP.NET using VB.NET or C# to the database with ADO.NET for Deleting Data
14. Create a page in ASP.NET using VB.NET or C# to the database with ADO.NET for Search Data.
15. Create a page in ASP.NET using VB.NET or C# to create a proxy.
16. Create a page in ASP.NET using VB.NET or C# that has a form taking the user"s name as input. Store this name in a permanent cookie & whenever the page is opened again, then value of the name field should be attached with the cookie"s content
17. Create a page in ASP.NET using VB.NET or C# to run video
18. Create a page in ASP.NET using VB.NET or C# to delete all cookies of your web site that has created on the client's computer.
19. Create a page in ASP.NET using VB.NET or C# to the database with ADO.NET for Inserting Data.
20. Create a page in ASP.NET using VB.NET or C# to the database with ADO.NET for Updating Data.

Paper – II

Course Code – **Course Name – Information Security & Cyber Law**

Learning Outcome

LO1	Given information on information security and threats students will be able to **understand** structure, mechanics and evolution of various crime threats and able to **remember** the security mechanism.
LO2	Given information on various security mechanism, students will be able to **define** various security tools used to protect the data
LO3	Given information on IT Act 2000 students will be able to **illustrate** different terminologies used in IT Act 2000
LO4	Given information on various tools used in security, students will be able to **recognize** which tool is best suited in field.

UNIT - I

- Information Security : Overview, need for information security, objectives of Information security. - Global information systems and their evolution, basics of information systems, role of the Internet and the World Wide Web. - Understanding about the threats to information systems security Building blocks of InfoSec, How Organizations manage security of their information systems Information security risk analysis fundamentals. - Importance of physical security and biometrics controls for protecting information systems assets. - Security considerations for the mobile work force. - Network security perspectives, networking and digital communications (overview only), security of wireless networks.

UNIT - II

- Cryptographic techniques and Encryption, Intrusion Detection Systems and Firewalls,security of virtual private networks. - Security issues in application development with emphasis on integration of enterprise applications, database security, operating security and security of electronic mailing systems. - Security models and frameworks and standards through introduction to the ISO 27001, SSE-CMM (systems security engineering – capability maturity model), COBIT (Control Objectives for Information and related technologies) and the SarbanesOxley Act (SOX) and SAS 70 (statement on auditing standards). - Privacy Fundamentals, business practice's impact on data privacy, technological impact on data privacy, privacy issues in web services and applications based on web services. - Information security best practices – staffing, audits, and disaster recovery planning and business continuity planning and asset management. - Ethical issues and intellectual property concerns for information security professionals – copy right, data protection etc. matters.

UNIT – III

- Introduction of IT Act 2000, main features of IT Act 2000, Digital Signature. - Access Control : Operating system Access Controls, Group and Roles, Access Control lists, Unix Operating System Security, Windows NT, Capabilities, Added Features in Windows 2000, Granularity, Sandboxing and Proof-carrying code,Hardware protection, Other technical Attacks. - Cryptography & PKI : Symmetric Cryptography, Asymmetric Cryptography, Keys, Hash Functions, Digital Signatures. - Distributed Systems - Concurrency, Fault Tolerance and Fault Recovery, Naming.

UNIT – IV

- Multilevel and Multilateral Security : Multilevel Security, Multilateral Security. - Electronic Banking –Banking and Bookkeeping. - Monitoring Systems –Introduction, Alarms, Prepayment Masters. - Biometrics : Physiological biometric techniques, behavioral biometric techniques, - New biometric techniques, biometric systems. - Incident Response : Incident Response, Prerequistes to planning an IRT. - Network attack and Defence : Most Common Attacks, Scripts Kiddies and Packaged Defence. - Management Issues : Organisational Issues, - Protecting E-commerce Systems – Introduction - Hacking – Introduction

Books Recommended

1. Information Systems Security Management - Nina S. Godbole (Wiley India Pvt. Ltd.)
2. Security Engineering - Ross Anderson
3. Information Security Management Handbook - Harold Tpton & Micki Krause (Auerbach Publications)
4. Network Security Essentials: Applications and Standards - W. Stallings (Pearson Education)
5. eSecurity and You - Sandeep Oberoi (Tata McGraw-Hill)
6. Cyber Laws – Singh Yatindra
7. Cyber Crime – Bansal S K
8. Cyber law , E-commerce & M-Commerce – Ahmand Tabrez
9. Handbook of Cyber and E-commerce laws – Bakshi P M & Suri R K
10. Management Fundamentals and Information Systems Dr. Sushila Madan (Taxmann's)

Paper – III

Course Code – **Course Name – Practical Android Programming**

Learning Outcome

LO1	Given information on basics interface and architecture student will able to **develop** and **grasp** of the Android OS architecture (using various android views and view groups).

LO2	Given information on designing different themes for android application which help Students will able to **Understand** the handling the data by using external devices and also for the networking communication application.
LO3	Given information will help the students to **understand** the geographical locations on the maps with the help of geo-coding and reverse geo-coding as well as– application will enrich with use of graphics and animation.
LO4	Given information will help Students to **Familiarize** with Android development by selecting tools for including device emulator, profiling tools and IDE as well as **Identity, analyze** data storage, retrieval, user preferences, files and content providers

UNIT - I

Getting an Overview of Android Introducing Android - Listing the Version History of Android Platform, Discussing Android APIs, Describing the Android Architecture Application Framework, Exploring the Features of Android, Discussing about Android Applications, The Application Components, The Manifest File, The Command-Line Tools, Developing and Executing the First Android Application, Using Eclipse IDE to Create an Application, Running Your Application, Exploring the Application, Using Command-Line Tools. **Using Activities, Fragments and Intents in Android -** Working with Activities, Creating an Activity, Starting an Activity, Managing the Lifecycle of an Activity, Applying Themes and Styles to an Activity, Displaying a Dialog in the Activity, Hiding the Title of the Activity, Using Intents, Exploring Intent Objects, Exploring Intent Resolution, Exploring Intent Filters, Resolving Intent Filter Collision, Linking the Activities Using Intent, Fragments, Fragment Implementation, Finding Fragments, Adding, Removing, and Replacing Fragments, Finding Activity Using Fragment, Using the Intent Object to Invoke Built-in Application. **Working with the User Interface Using Views and ViewGroups -** Working with View Groups, The LinearLayout Layout, The RelativeLayout Layout, The ScrollView Layout, The TableLayout Layout, The FrameLayout Layout, The TabLayout Using the Action Bar, Working with Views, Using the TextView, Using the EditText View, Using the Button View, Using the RadioButton View, Using the CheckBox View, Using the ImageButton View, Using the ToggleButton View, Using the RatingBar View, Binding Data with the AdapterView Class, Using the ListView Class, Spinner, Using the Gallery View, Designing the AutoTextCompleteView, Implementing Screen Orientation, Anchoring the Views of the Current Activity, Customizing the Size and

Position of the Views, Designing the Views Programmatically, Handling UI Events, Handling User Interaction with Activities, Handling User Interaction with the Views, Specialized Fragments, ListFragment, DialogFragment, PreferenceFragment, Creating Menus The Options Menu The Context Menu The SubMenus.

UNIT - II

Handling Pictures and Menus with Views - Working with Image Views, Displaying Images in the Gallery View, Displaying Images in the Grid View, Using the ImageSwitcher View, Designing Context Menu for Image View, Using the AnalogClock and DigitalClock Views, Embedding Web Browser in an Activity, Notifying the User Creating the Toast Notification, Creating the Status Bar Notification, Creating the Dialog Notification. **Storing the Data Persistently -** Introducing the Data Storage Options, Using Preferences, Using the Internal Storage Exploring the Methods Used for Internal Storage, Developing an Application to Save User Data Persistently in File, Using the External Storage Exploring the Methods Used for External Storage, Developing Application to Save File in SD Card, Using the SQLite Database Creating the Database Helper Class, Creating the Layout and Main Activity Class, Creating the Layout and Activity for the Insert Operation, Creating the Layout and Activity to Search a Record, Creating the Activity Class to Fetch All Records, Creating the Layout and Activity for the Update Operation, Creating the Layout and Activity for the Delete Operation, Executing the Database Operations, Working with Content Providers, Exploring the android.provider Package, Creating User-Defined Content Provider, Consuming User- Defined Content Provider. **Emailing and Networking in Android -** Building an Application to Send Email, Networking in Android, Getting an Overview of Networking Fundamentals, Checking Network Availability, Accessing Web Services UsingHTTP Post, Accessing Web Services Using the GET Method, Working with Binary Data and Text Files, Consuming JSON Services, Sockets Programming.

UNIT - III

Working with Location Services and Maps - Working with Google Maps, Exploring Google Maps External Library, Creating an Application Using Google Maps Android API, Disabling the Zoom Control Button, Changing the Map Type, Displaying the Specific Location and Adding Markers, Handling Map Gestures Interaction, Getting the Current Location of a User, Working with Geocoding and Reverse Geocoding. **Working with Graphics and Animation**

- Working with Graphics, Drawing Graphics to Canvas, Using the Drawable Object, Referencing an Image File, Defining Drawable in XML, Using the Shape Drawable Object, Working with the Nine Patch Drawable Graphics, Understanding the Concept of Hardware Acceleration, Working with Animations, The Property Animation, View Animation Drawable Animation. Audio, Video and Camera - Role of Media Playback Using Media Player Media Formats Supported by Media Player, Preparing Audio for Playback, Preparing Video for Playback, Creating Application to Play Audio and Video Using MediaPlayer, Recording and Playing Sound, Use of Media Store Audio Recording Application, Creating a Sound Pool Using Camera for Taking Pictures, Creating Video RecordingApplication.

UNIT - IV

Threads and Services - Introducing Threads Worker Threads Using AsyncTask, Introducing Services Exploring Services Essentials, Understanding the Lifecycle of a Service, Exploring the Service Class, Introducing the Service Class, Creating a Bound Service. **Bluetooth, NFC and Wi-Fi -** Working with Bluetooth Exploring the Android Bluetooth APIs, Permissions Required to Access Bluetooth, Setting Up the Bluetooth for an Application, Identifying the Bluetooth-Enabled Devices, Querying the Paired Devices, Discovering Devices Creating an Application Using Bluetooth Functionality, Connecting the Devices Using Bluetooth for Data Transfer, Connecting as a Server Connecting as a Client Working with Bluetooth Low Energy, Working with NFC, Exploring the Basics of NFC, Developing an Application Using NFC, Working with Wi-Fi, Exploring the Wi-Fi APIs, Creating an Application Using Wi-Fi. **Telephony and SMS -** Handling Telephony Displaying Phone Information Application Receiving Phone Calls Application, Making Outgoing Phone Calls Application, Handling SMS Sending SMS Using SmsManager, Sending SMS Using Intent, Receiving SMS Using the BroadcastReceiver Object, Role of Default SMS Providers. Hardware Sensors - Introducing Sensors Exploring the Sensor Framework, Managing Various Sensor Configurations, Understanding the Sensor Coordinate System.

Text Book:

1. Rradeep Kothari, Android Application Development – Black Book, Dreamtech Press.

Reference Books:

2. Prasanna Kumar Dixit, Android, Vikas Publishing.
3. Dawn Griffiths & David Griffiths, Head First Android Development, Shroff Publishers.
4. Ed Burnette, Hello Android, Shroff Publishers.
5. Jerome DiMarzio, Android – A Programmer's Guide, McGraw-Hill.
6. Dave MacLean, Satya Komatineni, Grant Allen, Pro Android 5, Apress.
7. Reto Meier, Professional Android Application Development, Wiley.

Practical List of Android Programming

1. Create —Hello World! android application. That will display —Hello World! in the middle of the screen in the red color with white background.
2. Write an android application to understand Activity, Intenta. Create sample application with login module. (Check username and password) and on successful login, go to next screen. And on failing login, alert user using Toast. Also pass username to next screen.
3. Create an android application that will change color of the screen and change the font size of text view using xml.
4. Create login android application where you will have to validate EmailID (UserName). Till the username and password is not validated, login button should remain disabled.
5. Create and login android application as above. On successful login, open browser with any URL.
6. Create an android application that will pass some number to the next screen, and on the next screen that number of items should be display in the list.
7. Create an android application that will change color of the screen, based on selected options from the menu.
8. Create an android application that will display toast (Message) on specific interval of time.
9. Create a android background application that will open activity on specific time.
10. Create an android application that will have spinner with list of animation names. On selecting animation name, that animation should affect on the images displayed below.
11. Create an android UI such that, one screen have list of all the types of cars. and on selecting of any car name, next screen should show Car details like : name, launched date, company name, images(using gallery) if available, show different colors in which it is available.

12. Write an android application to read phonebook contacts using content providers and display in list.
13. Write an android application to read messages from the mobile and display it on the screen.
14. Create an android application to call specific entered number by user in the EditText
15. Create an android application that will create database with table of User credential.
16. Create an android application to read file from asset folder and copy it in memory card.
17. Create an android application that will play a media file from the memory card.
18. Create an android application to make Insert, update, Delete and retrieve operation on the database.
19. Create an android application to read file from the sdcard and display that file content to the screen.
20. Create an android application to draw line on the screen as user drag his finger.
21. Create an android application to send message between two emulators.
22. Create an android application to take picture using native application.
23. Create an android application to pick up any image from the native application gallery and display it on the screen.
24. Create an android application to open any URL inside the application and clicking on any link from that URl should not open Native browser but that URL should open the same screen.
25. Create an android application that will create database with table of User credential.

Paper – 1II

Course Code – **Course Name – Angular JS**

Learning Outcome

LO1	Given information on basics of Angular JS student will be able to **design** the page and **apply** in real time applications
LO2	Given information on directives, controllers and modules student will be able to **Create** and **Use** while design their web page or web application

LO3	Given information on Forms and Dependency, students will be able to **design** Form with all validations and be able to **create** services.
LO4	Given information on application building student will be able to **create** single page application with animations

UNIT - I

Angular JS Basics - What is Angular JS? Why Angular JS? Why MVC matters, MVC-The Angular JS way Features of Angular JS, Model-View-Controller, My First Angular JS app.

Angular Expressions - All about Angular Expressions, How to use expressions, Angular vs JavaScript

Filters - Built-In Filters, Using Angular JS Filters, Creating Custom Filters

UNIT - II

Directives - Introduction to Directives, Directive Lifecycle, Binding controls to data, Matching directives, Using Angular JS built-in directives, Creating a custom directive

Controllers - Role of a Controller, Controllers & Modules, Attaching Properties and functions to scope, Nested Controllers, Using Filters in Controllers, Controllers in External Files

Angular JS Modules - Introduction to Angular JS Modules, Bootstrapping Angular JS

UNIT - III

Angular JS Forms - Working with Angular Forms, Model Binding, Forms Events, Updating Models with a Twist, Form Controller, Validating Angular Forms, $error object

Scope - What is scope, Scope Lifecycle, Scope Inheritance, Scope & Controllers, Root scope, Scope Broadcasting, Two-way data binding, Scope Inheritance, Scope & Directives, $apply and $watch, Scope Events

Dependency Injection & Services - What is Dependency Injection, Creating Services, Factory, Service & Provider, Using Dependency Injection, What are services, Using Angular JS built in services

UNIT - IV

Single Page Application (SPA) - What is SPA, Pros and Cons of SPA, Passing Parameters, Changing location, Installing the ng Route module, Configure routes, Resolving promises, creating a Single Page Apps

Angular JS Animation- Animate Module, CSS Transforms, CSS Transitions, Applying Animations

Reference Books

1. The Complete Guide to Angular Paperback – 6 February 2018 by Felipe Coury, Ari Lerner and Carlos Taborda
2. Angular in Action Paperback – Import, 2 April 2018 by Jeremy Wilken
3. Angular: From Theory To Practice: Build the web applications of tomorrow using the Angular web framework from Google. Kindle Edition by Asim Hussain
4. Angular 6 for Enterprise-Ready Web Applications: Deliver production-ready and cloud-scale Angular web apps 1st Edition, Kindle Edition by Doguhan Uluca
5. Angular: Up and Running: Learning Angular, Step by Step 1st Edition, Kindle Edition by Shyam Seshadri
6. Pro AngularJS (Expert's Voice in Web Development) Paperback – 7 April 2014 by Adam Freeman
7. Angular Development with TypeScript Paperback – 17 December 2018 by Yakov Fain and Anton Moiseev

Practical List of Angular JS

1. Design Order Form with a total price updated in real time, which contains name of five products and their prices. Create a bill amount for all the products, calculate GST on the billing amount, and display total amount.
2. Implement Angular JS to create your Resume.

3. Use Practical No.01 and initialize prices to 0 (zero) when form loads. (Use module, controller & directive).
4. Design a webpage which takes one number as an input and generate its factorial number (use module, controller).
5. Design a webpage, which takes inputs product name, product quantity and price. Generate table of entered values. When user clicks on table column title, it should sort that column values. (Use filter, array).
6. Design a webpage which display product name and product price using AngularJS $http Service from database. Display the content in tabular format.
7. Write a program to create notepad application using Angular JS
8. Write a program in Angular JS to create Navigation Menu that highlights the selected entry.
9. Write a program in Angular JS to create a simple inline editor - clicking a paragraph will show a tooltip with a text field.
10. Write a program in Angular JS to design an order form with a total price updated in real time, using filters.
11. Write a program in Angular JS to filter a list of items by typing into a text field
12. Write a program in Angular JS to change name using JS Controller.
13. Write a program in AngularJS to enter the text in the input field. & Angular JS will display an error message if the entered text is invalid.
14. Write a program in Angular JS to hide a Div by adding checkbox.
15. Write a program in Angular JS to print "Hello World"
16. Write a program in Angular JS to design cost calculator.
17. Write a program in Angular JS to change the color of input box by changing its value.